

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

Adaptive Bandwidth Throttling for Network Services

Inventor(s):

Murali R. Krishnan

ATTORNEY'S DOCKET NO. MS1-327USC1

RELATED APPLICATIONS

This is a continuation of U.S. Patent Application Serial No. 08/919,633, filed August 28, 1997, which is a continuation-in-part of U.S. Patent Application Serial No. 08/674,684, filed July 2, 1996, now U.S. Patent No. 5,799,002.

TECHNICAL FIELD

This invention relates to network servers resident on a host computer system and, in particular, to a bandwidth management system which throttles the demands by client processes executing on remote computer systems for network transmission bandwidth.

BACKGROUND

A computer network system has one or more host network servers connected to serve data to one or more client computers over a network. Fig. 1 shows a simple computer network system 20 with a single host network server 22 connected to multiple clients 24(1), 24(2), ..., 24(N) via a network 26. The clients 24(1)-24(N) send requests for data and/or services to the server 22 over the network 26. For discussion purposes, suppose the server 22 is configured as an Internet service provider, or "ISP". The ISP server 22 provides an email service 28 that handles electronic mail messages over the Internet 26 and a web service 30 that supports a web site accessible by the clients.

The network 26 is a medium with a predefined bandwidth capacity that is shared among the clients 24(1)-24(N). The network 26 is represented in Fig. 1 as a network pipeline to indicate a finite bandwidth capacity. The network 26 is representative of different network technologies (e.g., Ethernet, satellite, modem-based, etc.) and different configurations, including a LAN (local area network), a

1 WAN (wide area network), and the Internet. The bandwidth capacity depends on
2 the technology and configuration employed. For this example, suppose the
3 network 26 has a total bandwidth capacity of 1,000 kilobits per second (Kb/s).
4 Given this fixed bandwidth, the ISP administrator can allocate portions of the
5 bandwidth for the various services 28 and 30. For instance, the ISP administrator
6 might allocate 400 Kb/s to the email service 28 and 600 Kb/s to the web service
7 30.

8 As the clients 24(1)-24(N) access the services 28 and 30, they consume
9 bandwidth on the network 26. The responses from the host server 22 also
10 consumer bandwidth. When the allocated bandwidth for a service becomes
11 saturated with client requests and server responses (such as the web service when
12 bandwidth consumption reaches 600 Kb/s), some of the requests are either delayed
13 in transmission or not delivered to the intended destination. Therefore, some form
14 of request throttling mechanism is necessary to minimize network congestion and
15 efficiently utilize the allocated network bandwidth.

16 In the case of multiple network servers or services executing on a single
17 host computer system and sharing a fixed bandwidth communication link to the
18 network, some network servers can disproportionately allocate this network
19 bandwidth to their tasks, thereby excluding other concurrently executing network
20 servers from performing their requested operations. In this case, the bandwidth
21 throttling must be effected among the plurality of network servers which are
22 concurrently executing on the host computer system.

23 It is therefore a problem to allocate bandwidth to the network server
24 processes in a manner which enables the maximum number of requests to be
25

1 served without network congestion and to also avoid impacting other network
2 servers which may be executing on the same host computer system.

3 There have been many implementations of bandwidth allocation and
4 congestion control schemes to address this problem. U.S. Patent No. 4,914,650
5 discloses an integrated voice and data network which includes a multiplexer which
6 functions to connect the host computer system with the network. The multiplexer
7 is equipped with a voice queue for storing voice packets and a data queue for
8 storing data packets. Both the voice packets and the data packets are transmitted
9 uninterrupted for a respective predetermined interval, whose respective durations
10 may be different. Signaling messages which are exchanged among the computer
11 systems via the network preempt the voice and data transmissions to ensure that
12 signaling messages are serviced with very low delay and zero packet loss. in
13 addition, the bandwidth allocated for each type of transmission, if unused, can be
14 momentarily allocated to the other type of transmission to maintain a high level of
15 service.

16 U.S. Patent No. 5,313,454 discloses a feedback control system for
17 congestion prevention in a packet switching network. Congestion control is
18 achieved by controlling the transmission rate of bursty traffic when delay sensitive
19 data is present for transmission. The bursty data is relatively insensitive to delay
20 and can be queued for a reasonable period of time. Data indicative of the queue
21 length is broadcast via the network to the destination node where it is processed
22 and a control signal returned to the originating node to regulate the rate of
23 transmission of the bursty data.

24 U.S. Patent No. 5,359,320 discloses a scheduling mechanism for a network
25 arbitration circuit in a broadcast network environment. The scheduling

1 mechanism delays the arbitration circuit from seeking access to the network if the
2 network traffic exceeds a first predetermined threshold and the local traffic in the
3 node exceeds a second predetermined threshold. This scheduling mechanism
4 therefore responds to both local and global congestion to throttle the production of
5 new requests.

6 U.S. Patent No. 5,432,787 discloses a packet switching system which
7 appends a parity packet to each predetermined number of data packets. The
8 number of data packets which are transmitted before the parity packet is appended
9 thereto is a function of the network traffic and the measured network error rate.

10 U.S. Patent No. 5,477,542 discloses a packet switching network which
11 interconnects a plurality of terminal stations for transmitting video and voice data
12 packets. The terminal stations which are operating in the receive mode transmit
13 control signals to the associated transmitting terminal stations to indicate the
14 amount of delay that the received packets have experienced in traversing the
15 network. If the delay exceeds a predetermined threshold, the video packets are
16 delayed and the voice packets are preferentially transmitted, since the voice
17 packets are more sensitive to transmission delays.

18 Thus, there are numerous existing network congestion control mechanisms
19 available to regulate the transmission rate of data through a network. However,
20 the common thread in all of these systems is that a single control mechanism is
21 provided to effect the desired congestion control. These control schemes are
22 typically binary in nature, being either active or disabled. There is presently no
23 known hierarchical network congestion control system which differentially
24 responds to various levels of congestion. Furthermore, these congestion control
25

1 schemes operate without regard for the nature of the processes that are extant on
2 the network servers.

3 Additionally, Fig. 2 shows an example in which the ISP 22 supports
4 multiple domains 32(1)-32(M) on the same web service 30. For instance, it is not
5 uncommon for an ISP to support thousands of domains on the same web service.
6 To the client, however, each domain functions as its own service as if running on
7 its own HTTP (Hypertext Transfer Protocol) server on its own machine. Hence,
8 the ISP 22 is effectively running multiple "virtual services" on multiple "virtual"
9 HTTP servers, all from the same web service on the same machine. In such cases,
10 network bandwidth control cannot be limited to applying globally to all the virtual
11 servers. The all-or-nothing approach is unacceptable because the administrator
12 often desires to designate some virtual services as more or less critical than others.

13 14 SUMMARY

15 An adaptive bandwidth throttling system of the present invention provides a
16 hierarchically organized response to network congestion to escalate the actions
17 taken to mitigate the traffic presented to the network in response to various levels
18 of congestion. The bandwidth throttling system operates on a host computer
19 system to allocate bandwidth to the network servers which are executing on the
20 host computer system as a function of system administrator defined thresholds.
21 The management of the plurality of network servers can be independent of each
22 other, or may be coordinated, as the system administrator deems appropriate. In
23 addition, the bandwidth throttling system can customize the system response as a
24 function of the specific network process which is being regulated. Thus, the
25

1 throttling mechanism can be crafted to correspond to unique needs of the various
2 network servers.

3 The bandwidth throttling system implements a graceful diminution of
4 services to the clients by implementing a series (at least two) of successively
5 significant bandwidth throttling actions in response to a corresponding one of a
6 plurality of thresholds of increasing magnitude being exceeded. For example, the
7 bandwidth throttling system can delay a first class of services provided by a
8 network server in response to the effective bandwidth utilized by this network
9 server exceeding a first threshold. If the demand for the bandwidth by this
10 network server exceeds a second threshold, the bandwidth throttling system
11 escalates the throttling response and blocks the first class of services from
12 execution and can also concurrently delay execution of a second class of services.
13 The second level of response alternatively can include blocking only selected
14 members of the first class of services and delaying additional services, previously
15 not impacted by the bandwidth throttling process. The implementation of the
16 throttling process can be varied, to include additional levels of response (>2) or
17 finer gradations of the response, and to include subsets of a class of services. In
18 addition, the threshold levels of bandwidth used to trigger the throttling response
19 can be selected as desired by the system administrator. Typically, once the
20 effective bandwidth utilization is approximately equal to the allocated bandwidth
21 for the network server, the first level of the hierarchical bandwidth throttling is
22 activated. The second level of the hierarchical bandwidth throttling is activated
23 once the effective bandwidth utilization exceeds the allocated bandwidth for the
24 network server by greater than a predetermined amount.
25

1 By implementing a hierarchical response to excessive network traffic, the
2 impact on the various network services are minimized. The ability to customize
3 the bandwidth throttling to specific subclasses of services enables the system to
4 impact the services which are deemed by the system administrator to be of the
5 lowest priority and/or whose reduction of service levels has the most beneficial
6 effect on the network. Thus, the adaptive bandwidth throttling system provides a
7 graduated throttling process to incrementally reduce the demand for network
8 bandwidth without disrupting the provision of desired network services.

9 10 **BRIEF DESCRIPTION OF THE DRAWINGS**

11 Fig. 1 is a diagrammatic illustration of a host computer network system,
12 which is used to illustrate the present state of the art.

13 Fig. 2 is a diagrammatic illustration of the host computer network system
14 implemented with multiple "virtual servers" supported by a single service on a
15 single server machine.

16 Figure 3 illustrates in block diagram form the overall architecture of the
17 adaptive bandwidth throttling system and an environment in which it operates;

18 Figure 4 illustrates in flow diagram form the operation of the overall
19 system, including the adaptive bandwidth throttling system, in responding to
20 service requests;

21 Figure 5 illustrates in flow diagram form the operation of the adaptive
22 bandwidth throttling system.

23 Fig. 6 is a diagrammatic illustration of a host computer network system
24 having a server implemented with a bandwidth throttling system of this invention.
25

1 Fig. 7 is a flow diagram showing steps in a method for initializing the
2 bandwidth throttling system.

3 Fig. 8 is a diagrammatic illustration of a bandwidth throttling object that is
4 stored at the server and utilized by the bandwidth throttling system to track
5 bandwidth performance of individual virtual services.

6 Fig. 9 is a flow diagram showing steps in a method for handling client
7 requests at the ISP server.

8 Fig. 10 is a flow diagram showing steps in a method for processing a read
9 operation directed to a virtual service.

10 Fig. 11 is a diagrammatic illustration of a throttling strategy which uses a
11 threshold and offset value to establish a tiered approach to invoking various sets of
12 throttling actions depending upon the I/O activity.

13 Fig. 12 is a flow diagram showing steps in a method for updating
14 bandwidth measurements in individual BT objects.

15 Fig. 13 is a diagrammatic illustration of a histogram stored in a BT object to
16 keep statistics on I/O activity for the virtual service.

17 Fig. 14 is a flow diagram showing steps in a method for halting operation
18 of the bandwidth throttling system.

19 20 **DETAILED DESCRIPTION**

21 Network servers are processes which execute on a host computer system
22 and which function to serve requests for service received from remote computer
23 systems. The host computer system providing the requested service is typically
24 termed the server. The remote computer system initiating the request is termed the
25 client. The data exchanged between the host and remote computer systems is

1 transmitted in units termed packets, each of which consists of at least one byte of
2 data. The allocation of work between the client and server typically comprises a
3 client process requesting a server process to read and write identified data files.
4 The data files as well as the request and response messages are transmitted via the
5 medium of the network which interconnects the computer systems on which the
6 client and server processes execute.

7 The processing of a client originated service request begins with the
8 network server receiving one or more request packets from the client via the
9 network. Upon receipt of the request, the network server parses the request
10 contained in the request packets, processes the request and responds by
11 transmitting one or more response packets to the client via the network. If the
12 network becomes saturated with data transmissions, some of the request and/or
13 response packets may not be delivered to their intended destination. It is possible
14 that a host computer may be running a plurality of network servers, thus limiting
15 the processing resources which are available for any of the plurality of network
16 servers. In such a scenario, control of the amount of the network bandwidth used
17 by any one network server allows both that server as well as the other servers to
18 perform more efficiently.

19 System administrators manage the network servers by utilizing the
20 information that the administrators have collected regarding the processing
21 requirements of the various services available from the network servers, as well as
22 the requirements of the typical clients. In particular, some network services
23 generate data which is intolerant of transmission delays while other network
24 services generate data which is relatively delay insensitive. For example, bursty
25 data is relatively insensitive to delay and can be queued for a reasonable period of

1 time. In a combined voice-video data transmission system, video packets can be
2 delayed while the voice packets are preferentially transmitted, since the voice
3 packets are more sensitive to transmission delays than the video packets. In
4 addition, some services are more response critical, having a higher priority than
5 others. Using this data, as well as information regarding the data transmission
6 patterns of existing systems, the administrator can specify resource usage for the
7 plurality of network servers extant on the host computer system.

8 Figure 3 illustrates in block diagram form the overall architecture of the
9 adaptive bandwidth throttling system BT and an environment in which it operates,
10 while Figures 4 and 5 illustrate in flow diagram form the operation of the adaptive
11 bandwidth throttling system BT. The adaptive bandwidth throttling system of the
12 present invention is described as implemented in software, although this system
13 can alternatively be implemented as hardware elements or a combination of
14 hardware and software elements. The adaptive bandwidth throttling system
15 functions to limit the bandwidth usage of each of the plurality of services provided
16 by the plurality of network servers NS1-NSm extant on the host computer system
17 P to the allocated maximum network bandwidth. This control is architected to
18 achieve the minimum impact on the network servers NS1-NSm while concurrently
19 having the maximum impact on network congestion. In selecting an
20 implementation of a bandwidth control mechanism, it is important to note that
21 once a network service initiates a response process, that effort is lost if the
22 response is not executed to completion. Therefore, bandwidth throttling
23 procedures should terminate a service before substantial processing effort is
24 expended. In addition, the typical client-server interaction operates on a request-
25 response paradigm. In particular, the client transmits a request to the server and

1 the only communication that is received by the client is the response to the
2 request. There is no interprocess communication. Thus, rejecting and/or delaying
3 requests typically results in a subsequent retry by the requesting client process,
4 which consumes both processor and network resources, although in a time delayed
5 manner. The client process can continue to send requests to an overloaded server
6 without the server being capable of throttling this request process. An alternative
7 interprocess communication scheme enables the server to notify the client of the
8 server condition, thereby providing feedback to the requesting client to terminate
9 future requests until the overload is cleared. The adaptive bandwidth throttling
10 system of the present invention is operable in both of these environments.

11 The adaptive bandwidth throttling system BT is based on a feedback
12 system that continuously monitors the bandwidth consumed by each network
13 server NS1-NSm and initiates action when a network server reaches the threshold
14 defined by the allocated network bandwidth for that network server. In particular,
15 the bandwidth throttling system BT executes as a process on the host computer
16 system P. The host computer system P contains at least one and more typically a
17 plurality of network servers NS1-NSm which concurrently execute as independent
18 processes. The host computer system P is connected via an auxiliary function
19 driver AFD to a physical network N which interconnects the host computer system
20 P to one or more remote computer systems C1-Cn, each of which have operational
21 thereon a plurality of client processes (only client process CP is illustrated for the
22 sake of simplicity), each of which generate the service request packets. An
23 asynchronous thread queue ATQ interconnects the network servers NS1-NSm, the
24 bandwidth throttling system BT and the ancillary function driver AFD. The
25 asynchronous thread queue ATQ performs the input and output operations with

1 respect to the connected network N by providing functions to read, write and
2 transmit data files over network connections using the sockets capability of the
3 host computer system P. The asynchronous thread queue ATQ communicates
4 with the ancillary function driver AFD and the windows sockets driver (not
5 shown) resident on the host computer system P to perform the required input and
6 output operations over the network N.

7 Figure 4 illustrates, in flow diagram form, an example of the basic
8 operation of client-server communications in the context of the bandwidth
9 throttling system BT. At step 201, the client process CP resident on remote
10 computer system C1 generates a request for a network service, which network
11 service is provided by a network server NS1 which is executing on the host
12 computer system P. The generated request is processed at step 202 into a series of
13 request packets and transmitted via network N to the host computer system P
14 attached to network N. The request transmission is accomplished in a manner
15 which is well known, and the request can be addressed specifically to the network
16 server NS1 on host computer system P or the request can be addressed via use of a
17 mnemonic which identifies a service, which can be provided by any available one
18 of a plurality of host computer systems which are connected to the network N.
19 The service request transmitted over the network N is received from the client
20 process CP at step 203 by the ancillary function driver AFD and forwarded to the
21 asynchronous thread queue ATQ. The asynchronous thread queue ATQ queries
22 the bandwidth throttling system BT at step 204 to ensure that the requested
23 operation is permitted for the identified network service. This determination is
24 made at step 205 where the bandwidth throttling system BT retrieves the effective
25 bandwidth measure for the identified network server NS1 and compares this value

1 with data stored in a control table which is indicative of the network bandwidth
2 allocated to this network server NS1. At step 206, the bandwidth throttling system
3 BT transmits an indication of the determined action to be taken to the
4 asynchronous thread queue ATQ which processes the received request at step 207.

5 If the requested operation is permitted, the asynchronous thread queue ATQ
6 enables the operation to execute. If the requested operation is not permitted, the
7 asynchronous thread queue ATQ regulates the operation pursuant to the control
8 procedure indicated by the bandwidth throttling system BT. In particular,
9 operations that are designated as rejected are prevented from proceeding and a
10 control packet can be returned to the requesting client process CP via the network
11 N to indicate that the requested service is unavailable at this time. If the requested
12 operation is designated in the delay category, the asynchronous thread queue ATQ
13 stores the received request and returns a control packet to the requesting client
14 process CP to indicate that the operation is pending.

15 For every operation which is allowed to execute, the asynchronous thread
16 queue ATQ transmits data indicative of the number of bytes processed by
17 execution of the requested operation to the measurement subsystem MS of the
18 bandwidth throttling system BT. The measurement subsystem MS uses this
19 received data to update the bandwidth usage data stored in the bandwidth
20 throttling system BT and periodically computes the effective bandwidth for this
21 network server at regular intervals. The control subsystem CS of the bandwidth
22 throttling system BT uses the computed effective bandwidth to update its internal
23 control tables and thereby regulate the operation of the network servers NS1-NSm.

24 Figure 5 illustrates in flow diagram form the process used by the bandwidth
25 throttling system BT to regulate the operation of the various network servers NS1-

NSm extant on the host computer system P. The bandwidth throttling system BT monitors the operations performed by the asynchronous thread queue ATQ to ascertain the bandwidth utilized by each of the network servers NS1-NSm. The bandwidth throttling system BT consists of two subsystems: a measurement subsystem MS to measure the bandwidth usage for each of the network servers; a control subsystem CS which applies feedback based control to the asynchronous thread queue ATQ to limit the bandwidth used by each network server NS1-NSm. The network operations which are monitored by the bandwidth throttling system BT are: receive, send, and transmit file. The measurement subsystem MS, at step 301, monitors not only the operations which are performed by each network server NS1-NSm, but also the data flow rate for each operation, in the form of effective real time bandwidth consumed. The effective real time bandwidth is determined by calculating the bandwidth for each operation which is performed and averaging the bandwidth utilization over the last n operations performed. To limit the complexity, the monitoring subsystem MS does not maintain a complete history of all operations, but instead maintains a histogram of bandwidth values for the last most recent n time intervals. These values are accumulated by time stamping the start and end times of each operation. If the operation proceeds to completion, the monitoring subsystem MS calculates the bandwidth by dividing the bytes transferred during the operation by the time interval duration. The resultant bandwidth value is stored, at step 302, in the n last time interval histogram, which set of values is used at step 303 to periodically compute an effective bandwidth for this network server.

The control subsystem CS, at step 304, receives the effective bandwidth data generated by the measurement subsystem MS and uses this information to

1 regulate the operation of the various network servers. In operation, the control
2 subsystem CS invokes the measurement subsystem MS to compute the effective
3 bandwidth, M , for each network server. The control subsystem CS of the
4 bandwidth throttling system BT uses three classes of operations to characterize the
5 nature of the operation: Read (R), Write (W), and Transmit (T); in addition to two
6 subclasses: Large (L) and Small (S) to denote the size of the data involved. The
7 list of monitored operations is therefor: Read (R), Write- Small (WS), Write-
8 Large (WL), Transmit-Small (TS), and Transmit-Large (TL). The breakpoint
9 between large and small data transfers is empirically determined and can differ for
10 read and write operations, and can vary among the network servers NS1-NSm.
11 With these categories, the allocated bandwidth for a particular network server, and
12 the present effective bandwidth determined by the monitoring subsystem MS, the
13 control subsystem CS determines whether it is safe (allow), marginally safe
14 (delay) or unsafe (block) to perform a particular requested operation. This
15 decision is based upon a set for factors: the specific nature of the operation, the
16 dynamic behavior of the network server, the amount of processor and memory
17 resources consumed by and required by the requested operation, the estimated and
18 specified bandwidths. Thus, the operation of the adaptive bandwidth throttling
19 system can be customized for the operating characteristics of the specific network
20 server.

21 Tables A and B indicate two views of a typical list of the actions taken by
22 the control subsystem CS for each of the identified operations at various levels of
23 bandwidth consumption for one (NS1) of the plurality of network servers NS1-
24 NSm operational on the host computer system P:
25

Control Table - Table A

| | No Action Taken | Services Delayed | Services Blocked |
|---------------|-------------------|------------------|------------------|
| $M < B$ | R, WS, WL, TS, TL | | |
| $M \approx B$ | WS, WL, TS, TL | R | |
| $M > B$ | WS, TS | WL, TL | R |

Control Table - Table B

| | Action | Threshold |
|---------------|---|--------------|
| $M < B$ | Allow R, WS, WL, TS, TL | $B - \delta$ |
| $M \approx B$ | Allow WS, WL, TS, TL Delay R | |
| $M > B$ | Allow WS, TS Delay WL, TL Block R | $B + \delta$ |

In particular, it is assumed that the identified network server is allocated a maximum bandwidth of B , with the computed effective bandwidth consumed being M . The responses by the control subsystem CS are listed across the top of Table A and consist of: no action taken, services delayed, and services blocked. Table B provides an alternative presentation of the information provided in Table A. In the first case illustrated in Tables A and B, the effective bandwidth

1 consumed by the network server is less than the bandwidth (B) allocated for this
2 network server by greater than a predetermined amount $M > (\delta)$. In this instance, at
3 step 305, the control subsystem CS determines that the effective bandwidth does
4 not exceed the bandwidth (B) allocated for the identified network server NS1 and
5 no action need be taken, since there is sufficient bandwidth to perform all of the
6 requested operations. The control subsystem CS therefore takes no control action,
7 processing returns to step 301, and the measurement subsystem MS continues to
8 measure the bandwidth consumed by the network server and keeps track of the
9 effective bandwidth utilization.

10 In the second case listed in Tables A and B the first threshold $(B - \delta)$ is
11 exceeded, and it is determined at step 305 that the bandwidth consumed by the
12 network server is at or has begun to exceed the bandwidth (B) allocated for this
13 network server $(B - \delta < M < B + \delta)$. The control subsystem CS at step 306 determines
14 whether the effective bandwidth has also exceeded the second threshold $(B + \delta)$. If
15 not, control subsystem CS initiates the first level of the hierarchy of bandwidth
16 throttling actions to regulate bandwidth usage to avoid increased bandwidth
17 utilization at step 307 and processing then returns to step 301. This process
18 represents a substantially proactive response to avoid serious problems which may
19 be occasioned by inaction at this point in time. The control subsystem CS, in the
20 example illustrated in Tables A and B, functions to delay all read operations (R) to
21 limit bandwidth usage. Network servers receive requests from clients and act
22 upon them. Therefore, limiting the number of requests in the request queue for a
23 particular network server limits the bandwidth utilized by this network server.
24 The delay of read operations provides time for the request traffic to abate without
25 further action. This procedure "buys time" by delaying presently received read

1 requests for execution at a later time, in anticipation that the network traffic will
2 be at a reduced level as the delayed read operations are executed, thereby
3 "smoothing out" the request workload. This process anticipates that the traffic is
4 irregular and a peak load is simply a transient condition. In addition, the servicing
5 of a read request typically results in a subsequent write and/or transmit file
6 request, therefore delaying a read operation also further delays these subsequent
7 write and/or transmit file request, having a compound effect.

8 In the third case listed in Tables A and B, the first threshold ($B-\delta$) is
9 exceeded, and it is determined at step 305 that the bandwidth consumed by the
10 network server is at or has begun to exceed the bandwidth (B) allocated for this
11 network server. The control subsystem CS at step 306 then determines whether
12 the effective bandwidth has also exceeded the second threshold ($B+\delta$). If so, the
13 bandwidth utilized exceeds the bandwidth allocated for this network server(B) by
14 greater than a predetermined amount (δ) and further corrective measures must be
15 taken. Processing therefore advances to step 308 where the second tier of
16 bandwidth throttling is activated and processing then returns to step 301. A
17 significant impact on system performance is achieved by rejecting all read
18 requests (R) and delaying a subclass of the previously enabled write (WL) and
19 transmit (TL) requests. In particular, both large write requests (WL) and large
20 transmit requests (TL) are now delayed. The read requests (R) are typically
21 rejected by transmitting an indication to the requesting client that the server is
22 busy or the network is busy. Blocking (delaying) large write and transmit requests
23 delays their impact on the bandwidth and reduces bandwidth utilization quickly,
24 since the delay of a few of these requests has far greater impact than rejecting read
25 requests due to their processing and bandwidth intensive nature. The rejection of

1 large write and large transmit operations may be counterproductive, since a
2 significant amount of processing may have been expended when the bandwidth
3 throttling system BT initiates the delay control process, which expended resources
4 are recouped when the delay period is over. However, a further escalation of the
5 control subsystem CS operation (not shown in Tables A and B) can be the
6 rejection of large write and/or large transmit operations during a severe overload
7 condition.

8 Variations of the bandwidth throttling scheme illustrated in Tables A and B
9 are possible, and this implementation is provided for the purpose of illustrating the
10 hierarchical nature of the adaptive bandwidth throttling system BT and its
11 adaptability to accommodate the needs of a particular host computer system P and
12 the unique servers operational on the host computer system.

13 The adaptive bandwidth throttling system provides a hierarchically
14 organized response to network congestion to escalate the actions taken to mitigate
15 the traffic presented to the network in response to various levels of congestion.
16 The bandwidth throttling system operates on a host computer system to allocate
17 bandwidth to the network servers which are executing on the host computer
18 system as a function of system administrator defined thresholds. The management
19 of the plurality of network servers can be independent of each other, or may be
20 coordinated, as the system administrator deems appropriate. By implementing a
21 hierarchical response to excessive network traffic, the impact on the various
22 network services are minimized. The ability to customize the bandwidth throttling
23 to specific subclasses of services enable the system to impact the services which
24 are deemed by the system administrator to be of the lowest priority and/or whose
25 reduction of service levels has the most beneficial effect on the network. Thus, the

1 adaptive bandwidth throttling system provides a graduated throttling process to
2 incrementally reduce the demand for network bandwidth without disrupting the
3 provision of desired network services. In addition, the bandwidth throttling
4 system can customize the system response as a function of the specific network
5 process which is being regulated. The throttling mechanism can be crafted to
6 correspond to unique needs of the various network processes.

7 Fig. 6 shows the computer network system 40 having a host network server
8 42 connected to serve multiple clients 44(1), 44(2), ..., 44(N) over a network 46.
9 The network 46 is representative of many diverse network technologies (e.g.,
10 Ethernet, satellite, modem-based, etc.) and different configurations, including a
11 LAN (local area network), a WAN (wide area network), and the Internet. For
12 discussion purposes, the computer network system 40 is described in the context
13 of the Internet whereby the host network server 42 is an Internet Service Provider
14 (ISP) that provides services to the clients 44(1)-44(N) over the Internet 46. It is
15 noted, however, that this invention may be implemented in other networking
16 contexts, including LAN and WAN configurations.

17 The ISP network server 42 is connected to the Internet 46 via a data
18 transmission network connection that has a predetermined fixed bandwidth
19 capacity. The bandwidth is typically characterized in terms of kilobits per second
20 or "Kb/s". The clients 44(1)-44(N) share the bandwidth when accessing the
21 services provided by the ISP server 42.

22 The host network server 42 has a processing unit 50, a memory subsystem
23 52, and a display 54. The memory subsystem 52 includes both volatile memory
24 (e.g., RAM) and non-volatile memory (e.g., ROM, hard disk drive, floppy disk
25 drive, CD-ROM, etc.). The host network server 42 runs a network server

operating system 56. In the preferred implementation, the operating system 56 is the Windows NT server operating system from Microsoft Corporation, which is modified to incorporate the bandwidth throttling system described below. As one example implementation, the host network server 42 is a microprocessor-based personal computer configured with the Windows NT server operating system. It is noted, however, that other server configurations (e.g., workstation, minicomputer, etc.) and other operating systems (e.g., a UNIX-based operating system) can be used to implement aspects of this invention.

The host server 42 supports one or more services, as represented by services 58 and 62 (e.g., also referred to as network servers above). Two example services are an email service and a web service. Each service 58 and 62 presents itself to the clients as multiple “virtual services”, as represented by virtual services (VS) 60(1)-60(J) for service 58 and virtual services 64(1)-64(K) for service 62. Within the context of a web service, the virtual services correspond to different domains supported on the same web service. To the client, each domain appears as its own web service running on its own HTTP (Hypertext Transfer Protocol) server. In reality, the domain is simply one of many supported by the single web service on the same server. Hence, the host server 42 is said to support multiple “virtual services” or present multiple “virtual servers” using the same web service on the same machine.

Since the clients 44(1)-44(N) share the bandwidth capacity for the virtual services offered by the host server 42, there can be congestion at times whereby too many simultaneous client requests bombard the host server 42. To minimize congestion and promote efficiency, the host server 42 employs a bandwidth throttling (BT) system 70 that throttles requests in an effort to avoid bandwidth

1 saturation. The bandwidth throttling system 70 is shown implemented as a
2 software module incorporated into the operating system 56 as part of, for example,
3 the Internet Information Services (IIS) component in the operating system.
4 Alternatively, the BT system may reside as a separate component independent of
5 the operating system. It is further noted that the BT system 70 can be
6 implemented separately from the host server 42 to manage request traffic to the
7 services supported on host server 42, as well as services supported on other servers
8 (not shown).

9 The BT system 70 provides a global throttling approach that applies across
10 all of the services 58 and 62 supported by the host server 42. The global throttling
11 technique imposes successively significant bandwidth throttling actions in
12 response to increasing bandwidth consumption, as discussed above. For global
13 throttling, the administrator defines one or more global bandwidth thresholds that
14 must be surpassed to initiate some form of bandwidth throttling that applies to all
15 incoming requests.

16 This invention concerns an improvement of the global bandwidth throttling
17 system described in the above application. In addition to global throttling, the BT
18 system 70 enables a finer grain control of the bandwidth on a per virtual service
19 basis. That is, rather than applying global throttling control across all of the
20 services, the BT system 70 also permits throttling control at the virtual service
21 level. This empowers an ISP administrator to set and monitor different bandwidth
22 thresholds for individual virtual services, and to manage the flow of requests to
23 each virtual service independently of other virtual services.

24 The BT system 70 has a measuring subsystem 72 to measure the portion or
25 amount of fixed bandwidth that is being presently used by each of the virtual

1 services. In a preferred implementation, the control subsystem 72 tracks the
2 bandwidth utilization on a per virtual service basis using multiple bandwidth
3 throttling objects that are created to represent the virtual services. The bandwidth
4 throttling objects are described in more detail below with reference to Fig. 8.

5 The BT system 70 has a control subsystem 74 to facilitate a throttling
6 strategy that selectively throttles requests for the individual virtual services
7 independently of one another on a per virtual service basis. The control subsystem
8 74 applies throttling actions to individual virtual services depending upon the level
9 of bandwidth being consumed by that virtual service.

10 More particularly, the control subsystem 74 applies a first set of throttling
11 actions to requests for a particular virtual service, say virtual service 60(1), if the
12 presently used bandwidth measured for the particular virtual service 60(1) exceeds
13 a first threshold. These throttling actions may include allowing certain types of
14 requests (e.g., high priority requests) while delaying other types of requests (e.g.,
15 low priority requests), as prescribed by the administrator. The throttling actions
16 imposed on virtual service 60(1) are independent of any throttling actions that may
17 be imposed on other virtual services 60(2)-60(J) and 64(1)-64(K) so that only
18 requests bound for the virtual service 60(1) are affected by the actions.

19 If the I/O activity for the virtual service 60(1) continues to rise and the
20 bandwidth used by the virtual service 60(1) exceeds a second threshold, the
21 control subsystem 74 applies a second, more restrictive set of throttling actions to
22 the requests for that virtual service. In this case, the throttling actions may include
23 allowing only requests designated by the administrator as high priority, delaying
24 requests designated as medium priority, and rejecting requests designated as low
25 priority.

1 The BT system 70 has a born or "B" list 76 and an active or "A" list 78 to
2 help manage the bandwidth throttling objects. In general, the BT objects are
3 created for each virtual service. A pointer to a BT object is placed on the born list
4 76 when the BT object is created. When the virtual service is handling client
5 requests, the associated BT object is also placed on the active list 78 to indicate
6 that the virtual service is presently receiving or responding to requests. The active
7 list is thus a subset of the born list.

8 The purpose of keeping an active list is to prevent unnecessary bandwidth
9 calculations for BT objects that are not active. For an ISP that supports thousands
10 of domains, for example, it is anticipated that only a fraction of the domains (e.g.,
11 ten percent) will be active at any one time. The BT objects for frequently visited
12 web sites, such as ESPN® Sports Zone or MSNBC, might always be active,
13 whereas BT objects associated with rarely visited web sites are seldom active.
14 Only the BT objects on the active list are routinely updated as to their presently
15 used bandwidth. The BT objects on the born list, but not on the active list, are
16 passed over as the bandwidth calculations are unnecessary for these objects.

17 In the example implementation, the BT system 70 utilizes a asynchronous
18 thread queue (ATQ) support library 80 provided by the Windows NT operating
19 system to handle requests. The ATQ library 80 enables asynchronous input and
20 output operations.

21 Fig. 7 shows steps for initializing the BT system 70 in preparation for
22 handling client requests. This start sequence occurs when the operating system
23 and Internet Information Service (IIS) component is booted. At step 90, the IIS
24 reads a metabase maintained on non-volatile memory to obtain content data used
25 to construct the bandwidth throttling objects. The metabase contains data on any

1 virtual service that has previously registered with the operating system. The
2 metabase data includes the names of the virtual services, the bandwidth thresholds
3 for the virtual services, and the like.

4 With this information, the control subsystem 74 creates a BT object for
5 each virtual service (step 92 in Fig. 7). The BT object is stored at the server and
6 used to track the bandwidth performance of the associated virtual service. The
7 control subsystem 74 adds the BT object to the born list 76 (step 94 in Fig. 7).
8 The initial bandwidth measurement for the BT object is then set to null as no I/O
9 activity has yet taken place (step 96 in Fig. 7).

10 Fig. 8 shows a bandwidth throttling object 100 in more detail. The BT
11 object 100 has born and active fields 102 and 104 that facilitate placement of
12 pointers to the BT object 100 onto the born and active lists 76 and 78. The BT
13 object 100 further has a threshold data field 106 to hold a bandwidth threshold
14 specified by the administrator for the associated virtual service. The bandwidth
15 threshold 106 indicates a level of I/O activity for an associated virtual service that
16 is effective to trigger throttling actions on requests for the associated virtual
17 service. The thresholds are set by the administrator in a manner to avoid real or
18 potential congestion that may occur if no throttling action is taken.

19 The BT object 100 has a measured bandwidth data field 108 to hold a
20 measured bandwidth that is presently being used by the associated virtual service
21 to accommodate the I/O activity. The measuring subsystem 72 routinely measures
22 the bandwidth used by the virtual service and stores this value in the measured
23 bandwidth data field 108. One specific technique for determining the presently
24 consumed bandwidth is described below with reference to Figs. 8 and 9. This
25 technique involves statistical analysis using a histogram of I/O activity. More

1 particularly, the measuring subsystem counts the number of bytes passed to or
2 from the virtual service during fixed time intervals. To support this measurement
3 technique, the BT object 100 includes a histogram data field 110 to hold data
4 indicative of the I/O activity for the virtual service measured at fixed time
5 intervals. A counter 112 maintains a pointer to a memory cell in which the I/O
6 count for the current interval of the histogram is to be stored. A time field 114
7 keeps a time value that is used in calculating an average bandwidth consumption
8 over the multiple histogram intervals.

9 The BT object 100 further includes a blocked list data field 116 to hold a
10 collection of requests for the virtual service that have been temporarily delayed as
11 a result of a throttling action. The BT object 100 also keeps statistics 118 relevant
12 to the bandwidth maintenance. These statistics might include information such as
13 the I/O activity, when or how often thresholds are exceeded, when and what
14 throttling actions are imposed on the virtual service (i.e., how many requests for
15 the virtual service are being allowed, blocked, or rejected), and so forth. The
16 statistics 118 can be presented to the administrator on demand, and displayed in a
17 user interface on the display 54 to assist the administrator in analyzing
18 performance of individual virtual services.

19 The BT object 100 contains process code 120 for performing methods used
20 in the control and throttling of bandwidth. Table 1 shows an example set of
21 methods that might be contained in the BT object 100.

Table 1

| <u>Method</u> | <u>Description</u> |
|-------------------|---|
| SetThreshold | Set the bandwidth threshold for this object. When measured bandwidth exceeds this value, the control system 74 will take appropriate throttling action. |
| GetThreshold | Query the currently set bandwidth threshold. |
| UpdateBytesXfered | Called when bytes have been transferred and the transfer pertains to this bandwidth throttling object. |
| GetStatistics | Query bandwidth maintenance statistics. |
| UpdateBandwidth | Update the internally maintained bandwidth measurement for this object. |

Establishing individual BT objects 100 for each virtual service is advantageous because the administrator can set bandwidth thresholds for on a per virtual service basis. Thus, some virtual services might be allocated more bandwidth, or a higher priority of usage, than other virtual services. For example, suppose the ISP server supports three virtual services. Two of the virtual services pay the same amount for a basic web site, while the third virtual service pays a premium for a premier web site. The administrator might wish to set the threshold(s) for the premier virtual service at a higher level than the basic virtual service. In this manner, throttling actions will be activated first for the basic virtual services to control request traffic before any throttling action is initiated for the premier virtual service, thereby allowing the I/O activity to continue on the premier virtual service unimpeded.

1 Fig. 9 shows steps for handling client requests at the ISP server 42. At step
2 130, the ISP server 42 receives from a client a request for a virtual service
3 supported by the server. In the context of a web site, the request might be in the
4 form of a universal resource locator (URL), such as "http://www.microsoft.com/".
5 The ISP server 42 assigns an asynchronous thread Context (ATQ Context) from
6 the ATQ library 80 to handle the incoming request (step 132 in Fig. 9). The ATQ
7 Context interconnects the BT system with the appropriate virtual service. The
8 ATQ library also supports the I/O operations with respect to the network by
9 providing functions to read, write, and transmit data files over the network
10 connection using socket capability available at the server (necessary network
11 access data including socket are stored in the ATQ Context).

12 Part of the setup of the ATQ context is to establish a connection (step 134).
13 This entails specifying a connection callback that the ATQ library will call when a
14 request arrives. The ATQ passes completion information, status information, and a
15 special context value to the callback. All subsequent asynchronous I/O operations
16 between the client and ISP utilize the special context value to allow the ATQ
17 library to operate properly and independently of other ATQ contexts that may be
18 active.

19 The ISP server 42 parses the client request to identify the virtual service
20 sought by the request (step 136 in Fig. 9). The BT system 70 next determines
21 whether a BT object 100 exists for the virtual service (step 138). For this
22 determination, the BT system 70 checks the born list 76 to see if any BT object
23 has been created for the requested virtual service. If a BT object does not exist
24 (i.e., the "no" branch from decision step 138), the BT system 70 handles requests
25 for the virtual service using the global bandwidth throttling parameters (step 140).

1 Conversely, if a BT object exists for the virtual service (i.e., the “yes” branch from
2 decision step 138), the BT system 70 binds the client connection with the BT
3 object (step 142 in Fig. 9). The BT object remains bound to the client connection
4 until the client is finished with its requests.

5 For purpose of continuing discussion, assume that a BT object is located
6 and bound to the connection. Next, the ISP server processes the request (step 144
7 in Fig. 9). The request is characterized as one of three types: read, write, and
8 transmit file. A read request is one in which the server is waiting to read data from
9 the client (or the server seeks data from the client) . A write request seeks to write
10 data from the server to the client. A transmit file request asks the ISP to
11 download a document or file, such as occurs usually during an initial HTTP GET
12 operation.

13 Fig. 10 shows steps for processing a read request. It is noted that similar
14 steps are taken for a write request and a transmit file request. The ATQ uses the
15 context value and invokes the operating system to perform a read operation (step
16 150 in Fig. 10). The service sets up a callback function in the ATQ Context to be
17 called when the operation completes. The ATQ checks with the control subsystem
18 74 to determine whether the BT object associated with the requested virtual
19 service is on the active list 78 (steps 152 and 154). If not (i.e., the “no” branch
20 from step 154), the control subsystem 74 adds the BT object 100 to the active list
21 (step 156 in Fig. 10). On the other hand, if the BT object 100 is on the active list
22 78 (i.e., the “yes” branch from step 154), the control subsystem 74 updates the BT
23 object state (step 158 in Fig. 10).

24 At step 160, the control subsystem evaluates a current set of throttling
25 actions to determine whether the read operation for the requested virtual service

1 can be performed. The evaluation is based on the present I/O activity for the
2 virtual service as maintained in the BT object for that virtual service. More
3 specifically, the control subsystem 74 extracts the measured bandwidth 108 and
4 the bandwidth threshold 106 from the BT object 100 and compares the two values.
5 Different throttling actions are taken depending upon the comparison results.

6 The BT system 70 preferably employs an adaptive, hierarchical throttling
7 strategy. In one preferred technique, the administrator establishes threshold zones
8 based on the threshold T and an offset value δ above and below the threshold T
9 (i.e., $T \pm \delta$). The result is a three-zone control area subdivided by two thresholds
10 (i.e., $T - \delta$ and $T + \delta$).

11 Fig. 11 illustrates the tiered control strategy. The vertical axis represents
12 bandwidth usage, measured in terms of I/O activity as the number of bytes being
13 passed to or from a virtual object within a predefined timeframe. If the bandwidth
14 being used by the virtual service is less than the first threshold (i.e., the first zone),
15 no throttling actions are taken. If the bandwidth usage exceeds the first threshold
16 but is less than the second threshold (i.e., the second zone), a first set of throttling
17 actions is taken. If the bandwidth usage exceeds a second threshold higher than
18 the first threshold (i.e., the third zone), a second set of throttling actions is taken.

19 The different sets of throttling actions affect operations differently
20 depending upon a plan devised by the administrator. In setting the actions, the
21 administrator takes into account the nature of the operation, the dynamic behavior
22 of the network server, the amount of processor and memory resources consumed
23 by and required by the requested operation, and the estimated and specified
24 bandwidths. As one example, suppose the control subsystem characterizes all
25 operations as either read (R), write (W), or transmit (T). In addition, the control

subsystem provides two subclasses large (L) and small (S) to denote the size of the data involved in handling the three operations. Accordingly, the possible operations are read (R), write-small (WS), write-large (WL), transmit-small (TS), and transmit-large (TL).

In this example, there are three possible actions: allowing a request to proceed, rejecting the request, and blocking (i.e., delaying) the request until a later time. Table 2 shows a hierarchical, adaptive throttling strategy for this example.

Table 2

| <u>Measured v. Threshold</u> | <u>Action</u> |
|--|---|
| $M < T - \delta$ (First Zone) | Allow: R, WS, WL, TS, TL Delay: None Reject: None |
| $T - \delta \leq M \leq T + \delta$ (Second Zone) | Allow: WS, WL, TS, TL Delay: R Reject: None |
| $M > T + \delta$ (Third Zone) | Allow: WS, TS Delay: WL, TL Reject: R |

An electronic version of table 2 can be stored in the BT system 70 for use in determining a set of throttling actions to apply to incoming requests.

With reference again to Fig. 10, suppose the control subsystem 74 finds that the measured bandwidth utilized by the virtual service (as indicated by the measure bandwidth parameter 108 in BT object 100) exceeds the first threshold but not the second. The control subsystem 74 looks up in table 2 what throttling action (if any) is to be applied to a read operation for a virtual service whose

bandwidth exceeds the first threshold. In this case, the control subsystem 74 obtains from the table a throttling action in the form of a delay request. The control subsystem 74 informs the ATQ that the read operation is to be delayed temporarily until bandwidth usage for the requested virtual service decreases.

It is noted that the above table of throttling actions is universal and can be used by the control subsystem 74 for determining the fate of requests destined for any of the virtual services. However, the bandwidth thresholds and measured bandwidth parameters utilized by the control subsystem 74 during the table look-up are specific to the virtual services themselves and locally maintained in the BT objects associated with the virtual services.

After the I/O for the read operation completes, the ATQ callback is called. This callback is passed the completion status, which includes the number of bytes transferred in the operation and the error codes, if any errors occurred. This information is recorded in the BT object and the state of the BT object is updated.

Fig. 12 shows steps that are routinely performed to update the measured bandwidth parameter in all active BT objects. For each BT object on the active list 78 (step 170 in Fig. 12), the measuring subsystem 72 updates the measured bandwidth 108 (step 172). By considering only the BT objects on the active list, the BT system 70 reduces the amount of administrative computation time used to update the bandwidth values.

One specific technique for computing the bandwidth being consumed by the virtual service is to utilize a histogram. Fig. 13 shows a histogram having ten intervals 1-10 of fixed duration (e.g., one second each). Within each interval, the BT object tracks the I/O activity for the virtual service in terms of the total number of bytes. Byte information pertaining to each operation occurring within an

interval is passed to the BT object during the ATQ callback. Depending on the I/O activity, different byte counts are likely to occur in the various intervals, as represented in Fig. 13.

The byte counts are kept for each interval in the histogram data field 110 of BT object (Fig. 8). The data field has ten cells to maintain the counts of the ten intervals. This data field is implemented using a circular memory that permits a continuous cycle of the ten memory cells. A current counter 112 indicates which memory cell is presently being filled with byte count data. The BT object 100 also tracks the time consumed for the I/O activity to transfer the bytes to or from the virtual service. The time parameter is kept in data field 114 of the BT object 100 (Fig. 8).

To compute a measured bandwidth, the measuring subsystem 72 totals the byte counts contained in the histogram data field 110 and divides that result by the total time kept in field 114. This calculation yields an average bandwidth usage over the ten-interval time frame. The average bandwidth is then placed in the data field 108 for future use in determining throttling actions for the specific virtual service associated with the BT object.

With continuing reference to Fig. 12, the update process may discover that the virtual service has not recently received any I/O requests. At step 174, the measuring subsystem 72 determines whether any activity has occurred during a past period of preset duration (e.g., the time necessary to cycle through the histogram or longer). If there is activity (i.e., the “yes” branch from step 174), the BT object remains active and flow continues to the next BT object on the active list (step 178). On the other hand, if no activity has occurred (i.e., the “no” branch from step 174), the control subsystem 74 removes that BT object from the active

1 list 78 (step 176 in Fig. 12). The BT object remains on the born list 76, but is no
2 longer carried on the active list 78. Flow then continues to the next BT object on
3 the active list (step 178).

4 Fig. 14 shows steps in a process for ending the BT system 70. At step 180,
5 all existing I/O operations are permitted to complete. The measuring subsystem
6 72 and control subsystem 74 cleanup the BT objects (step 182) and remove the BT
7 objects from the active and born lists 76 and 78 (step 184).

8 The BT system described above is advantageous over prior art bandwidth
9 management techniques because it allows bandwidth control on a per virtual
10 server basis. As a result, the administrator is afforded maximum flexibility at
11 setting fine-tuned throttling policies that impact the virtual services independently
12 of one another.

13 Although the invention has been described in language specific to structural
14 features and/or methodological steps, it is to be understood that the invention
15 defined in the appended claims is not necessarily limited to the specific features or
16 steps described. Rather, the specific features and steps are disclosed as preferred
17 forms of implementing the claimed invention.